DoD Architecture Framework Version 2.0

The Essential DoDAF: A User's Guide to Architectural Description Development
May 2009

This page left intentionally blank

Table of Contents

This page left intentionally blank

1. INTRODUCTION

This volume introduces the Department of Defense Architecture Framework (DoDAF), Version 2.0. In the sections that follow, information, techniques, and tips on how to use DoDAF V2.0 to best advantage are presented with examples to help you create the architecture that best suits your needs. ***Note: this volume does not replace the DoDAF since it is only an abbreviated introduction to the Framework. "Essential DoDAF" is intended to be a non-technical description and tutorial on the architecture development process utilizing DoDAF V2.0. This volume should be used in conjunction with DoDAF for the best effect.***

DoDAF V2.0 is the latest revision of the Framework, which has been evolving since publication of the Command, Control, Communications, Computing, Intelligence, Security and Reconnaissance (C4ISR) Architecture Framework was published in 1997. The name was changed in 2000 to DoDAF with the publication of DoDAF V1.0. An interim version, DoDAF V1.5 was published in 2007.

If you are a new manager and/or team member who is working on your first architecture effort, use this volume as a tutorial. Experienced architectural team members can use this volume to refresh their knowledge, and to understand what has changed in Version 2.0. In each case where a major change has occurred, the paragraph will start with an arrow symbol that highlights the change.

In brief, there are a number of major changes to the architecture development process described by the three volumes of DoDAF V2.0 that significantly differs from previous versions. These are:

Volume 1:

- The major emphasis on architectural description development has changed from a product-centric process to an information-centric process designed to provide decision-making data organized as information for the manager.
- The three major views of architectural description described in previous version (e.g., Operational, Technical, and System) have been changed to more specific views that relate to the collection of architecture-related data that can be organized as useful information for the manager in decision-making.
- 'Products' have been replaced by 'models' and 'views' that are used to represent specific architectural data and derived information.
- The Department initiatives for Architecture Federation and Tiered Responsibility have been incorporated. Requirements for sharing of data and derived information in a Federated environment are described.
- Specific tiers of architecture within the Department have been identified and described (e.g., Department, Segment/Capability, Component and Solution).
- Linkages to the Federal Enterprise Architecture are defined and described.

- Architecture constructs originally described in the Ministry of Defense (UK) Architecture Framework (MODAF), the NATO Architecture Framework (NAF), and the Open Group Architecture Framework (TOGAF) are adopted for use within DoDAF.
- A New DoDAF Meta-model (DMM), containing Conceptual, Logical, and Physical Exchange Specification views has been created.
- Examples of presentation graphical representations (models) have been provided as examples to assist managers in determining how architectural data and other derived information can be visualized for use in decision-making.

Volume 2:

- For the architect, DoDAF V2.0 changes the focus of the Architecture Development Process, as described in Section 1.5, "What Does the Architect Need to Do?" The basis of the Architecture Development Process is now the Data Meta-model Groups, which are described in Section 2.
- With the focus on data, DoDAF V2.0 does not have "products" but has "models" and "views". Rather than the Operational View-5 (OV-5) Operational Activity Model, there is the "Activity Model" with the same supporting data. This shifts the focus of the architectural description development effort onto data early in the architectural description development process.
- In DoDAF V1.5 and previous versions, "Nodes" are logical concepts that caused issues in the exchange and discussion of architectures. The concrete concepts were incorporated into the DoDAF Meta-model. Since "Nodes" are logical concepts that could be used to represent the more concrete concepts of locations, facilities, systems, activities, organizations or combinations of those things, DoDAF V2.0 focuses on those concrete concepts. There will NOT be a mapping of "Node" to the DoDAF V2.0 Meta-model Groups, concepts, classes, or associations. For the Architect, there are some changes in architectural description development:
  - When appropriate, DoDAF V1.X architectures that use the "Node" concepts will need to be updated to express the newer, concrete concepts in place of the abstract concept represented by "Node". Otherwise, when pre-DoDAF V2.0 architectural descriptions are compared with DoDAF V2.0 architectural descriptions, the concepts that "Node" cannot be clearly mapped to the newer architectural description.
  - DoDAF V2.0 architectural descriptions will need to express the concrete concepts (locations, facilities, systems, activities, organizations, etc).

Volume 3: An entirely new volume, Volume 3 contains the Physical Exchange Specification (PES), useful to developers, and software vendors for ensuring that data created in their tools is sharable with others in the DoD community.

The former DoDAF *Deskbook* is now the DoDAF Journal, and is in electronic format found at: HTTP://www.us.army.mil/suite/page/454707

Two major themes thread though this volume as they do in DoDAF itself. First, DoDAF is data-centric and not product-centric as in previous versions. Thus, the majority of information contained here discusses how to effectively and efficiently collect the RIGHT DATA, for the RIGHT REQUIREMENT, in the RIGHT WAY, that meets the decision-makers needs.

Second, the concept of 'Fit for Purpose', which we discuss in depth in Section 2, encourages and facilitates the scoping and organization of an architectural description effort toward collecting only the data needed for the requirement, and only those models and views (which we used to call products) needed to represent the data for analysis and presentation.

A description of the steps to take for developing architectural descriptions under DoDAF V2.0 can be found in Section 7, Volume 1 of DoDAF V2.0.

What exactly is an architectural description?

The definition used for 'architecture' in DoDAF is *"A set of abstractions and models that simplify and communicate complex structures, processes, rules, and constraints to improve understanding, implementation, forecasting, and resourcing."*

An 'Architectural Description' is defined by international standard as *"A collection of information products used to document an architecture." (ISO/IEC WD1 42010 7 Apr 2007)*

In simpler terms, an architecture is a set of data collected about a process, system, or higher level construct, such an enterprise, that is graphically represented (i.e., 'described') so that non-technical users can understand the process or system, make decisions based on the data, forecast future event or need, and apply that information to resources (i.e., people, hardware, software, or other assets) to the need.

This version of DoDAF incorporates several international standards applicable to development of architectures and their accompanying architectural descriptions[1].

A note on terminology used in this document:

Models

Visualizing architectural data is accomplished through ***models*** (e.g., the 'products' described in previous versions of DoDAF). Models (Which can be documents, spreadsheets, dashboards, or other graphical representations) serve as a template for organizing and displaying data in a more easily understood format.

Views

When data is collected and presented in a model format, the result is called a ***view***.

DoDAF V2.0 discusses DoDAF-described Models and Fit-for-Purpose views:

- ***DoDAF-described Models (also referred to as Models)*** are created from the subset of data for a particular purpose and are fully explained in DoDAF, Volume 2. These views are useful as examples for presentation purposes, and can be used as described or modified as needed.

- ***Fit-for-Purpose Views*** are user-defined views of a subset of architectural data created for some specific purpose (i.e., "Fit-for-Purpose"). While these views are not described or defined in DoDAF, they can be created, as needed, to ensure that presentation of architectural data is easily understood within an agency. This enables agencies to use their own established presentation preferences in their deliberations.

Viewpoint

Organized collections of views (often representing processes, systems, services, standards, etc.) are referred to as ***viewpoints,*** and with appropriate definitions are collectively called the ***Architectural Description.***

The Views described in DoDAF, including those that are legacy views from previous versions of the Framework, are provided as pre-defined examples that can be used when developing presentations of architectural data. DoDAF does not prescribe any particular views, but instead concentrates on data as the necessary ingredient for architecture development. *However, other regulations and instructions from both the Department of Defense (DoD) and the Chairman, Joint Chiefs of Staff (CJCS) have particular presentation view requirements. These views are supported by DoDAF 2.0, and should be consulted for specific view requirements.*

[1]International Standards Organization (ISO) (2007*). Systems and Software Engineering – Architectural Description* ISO/IEC WD1 42010 7 Apr 2007.

2. DETERMINING THE INTENDED SCOPE AND USE OF THE ARCHITECTURE ("FIT FOR PURPOSE" ARCHITECTING) (STEP 1)

Establishing the purpose and scope for architecture development is critical to ensuring that its use is consistent with project goals and objectives established by the process owner or decision-maker. In DoDAF V2.0, we call this "***Fit for Purpose.***"

> ***"Fit-for-Purpose"*** *describes an architecture that is appropriately focused (i.e., responds to the stated goals and objectives of process owner) and meets those intended objectives, either to directly support customer needs, or improve the overall process undergoing change.*

2.1 Establishing Purpose/Use for an Architecture

Establishing the purpose for architecture development starts with requirements. Most managers are too busy to informally decide on a Monday morning that they want to create a team and develop architecture of any type. Developing architectures takes time away from other activities, and removes often critical employees from their normal duties, even if on a part-time or sporadic basis. For these reasons alone, architectural development most commonly falls into three categories:

- *Mandated architectures* – those architectures required for reporting on on-going or proposed project.
- *Developmental architectures* – those architectures that provide needed systems, services, or other technical information for updates, upgrades, or replacement of IT asset.
- *Process improvement architectures* – those architectures created to document business processes that are broken, need improvement, or are created to increase understanding of business operations.

When architecture is required or desired, and management is willing to provide resources for its development, the team created to support the architect in developing architecture should use their time for maximum benefit and value. The architecture should be created to carefully describe the problem or desired outcome, and the architecture effort then organized to meet that description. This section and those that follow will assist in meet that objective.

Developing an accurate description of the architecture (i.e., architectural description) needed to support the established requirement involves an understanding of several critical elements, such as:

- Stakeholder Requirements.
- Critical Issues that affect organizational mission.
- Established target objectives for business/tactical operations.
- Potential ways to evaluate progress and success.

Each of these needs to be addressed early in the development process in order to ensure that architecture development proceeds along the expected course, and is not diverted by issues that have less relevance or priority. It is especially important to seek out and understand stakeholder's needs—both internal and external stakeholders—and seek to address their concerns in any development effort. These concerns often reflect an understanding and appreciation of the critical issues facing an organization at any level that impact on mission requirements or

Architecture development, and production of the architectural description, must support management decision-making in the establishment of target business/operational objectives and/or be useful in revising those objectives to fit changing needs.

That leads to Step 1 in the DoDAF architecture development methodology—***Determine the intended use of the architecture***. A simple checklist provides the key questions needed to complete this step so that a firm foundation is built for the architecture development effort that can support accurate and useful architectural description models and views. The intent is answering the question *'What purpose/use is intended with this architecture development?'*

Step 1: Determine the intended use of the architecture.

Each of the sections of this guide has similar checklists that highlight critical information needed for successful architecture description creation. See **Figure 2.1-1**.

Checklist for Step 1: Purpose/Use

___Architecture name
___Purpose/use for the Architecture is Determined
___Determine Critical Issue(s) to be addressed/accomplished by the architecture development
___Establish Target objectives to be reached
___Define Significant External Requirements to be addressed (i.e., JCIDS, DAS, interoperability Documentation)
___Determine Key Stakeholders (Internal/External)
___Identify Potential Barriers to Success
___Develop Initial Evaluation Metrics

Figure 2.1-1: Checklist for Step 1: Purpose/Use

The *architecture name* should be descriptive of the purpose of the architecture (e.g., Defense Spectrum Allocation Management Architecture, or Army Real Property Asset Management Architecture), contain a version number and approval date. That type of naming convention enables a level of assurance for any subsequent user of the architecture that it is the correct version desired. DRAFT architecture descriptions should

be clearly labeled as such until final approval is obtained. The following examples show notional architecture names

Defense Spectrum Allocation Management Architecture, Version 1.0 (DRAFT) 21 January 2009

or

Army Real Property Asset Management Architecture, Version 1.0, 21 January 2009

Determining the purpose and use of an architecture is based on responses to two questions:

- Why is the architecture being created?
- When completed, how will the architecture be used?

Defining well the purpose and use of the architecture is critical to later steps in the methodology. Often, management has a requirement that involves creation of architecture to support a system or service development effort. However, as the architecture is defined, based on the requirement, it may become clear that this is a very large effort that must be done in phases, or stages, and may involve several inter-related processes, each of which may need their own architecture and accompanying architectural description. In addition, the architecture development effort may involve links to other existing architectures, data managed by separate Communities of Interest (COI), or data received or transmitted to external sources.

Included in the definition of architecture descriptions are the *critical issues to be addressed/accomplished through the development effort*. In architect terms, these are ***critical success factors***, and they relate to those 'things' that need to happen during the development that will meet or exceed management expectations. Examples of critical success factors could include:

- Create only those views needed to explain the purpose of the architecture and facilitate analysis of data.
- Provide clear definition and description of how the architecture will foster and advance net-centric operations.
- Utilize to the maximum data in common use in the Department, registered in the Defense Metadata Registry, and integrated across architectural description views.

Each architectural description statement of purpose should include *target objectives*, and *initial metrics designed to measure success* against those objectives. These objectives provide another means to evaluate conditions in later steps of the methodology, such as risk management and mitigation, and a myriad of potential analysis capabilities.

Two final aspects of determining purpose and use of an architectural description involve *identification of external requirements* that must be met by the development. These include JCIDs, DAS, Systems Engineering, Interoperability, Portfolio management, and other major Departmental Programs and requirements that require specific views of the architecture for review and compliance determination. More broadly, it is important in this early step of the methodology to identify any *potential stakeholder* who may be impacted by architecture development, and subsequent change that occurs. Early identification of stakeholders and other external requirements will aid in review and securing approval of the architecture once completed.

Taking the time up front to clearly establish WHAT is needed in an architecture and HOW the architecture will be used once created will greatly simply later stages of the development effort. Establishing the WHAT and HOW of the effort should be done in conjunction with defining an initial set of high-level success criteria for determining project progress and success.

2.2 Determining the Scope of an Architecture (Step 2)

Step 2 of the methodology described in DoDAF establishes the scope of the architecture effort.

Step 2: Determine the scope of the architecture

The scope of an architecture is a description of its boundaries. The boundaries of architecture are derived from the purpose and intended use, and describe:

- What type of architecture is to be developed.
- How that architecture links to, or intersects with, other architectures.
- How the architecture will assist in fulfilling the need described in the requirement statement.

The first step in determining scope is to evaluate which of the major DoD decision-making activities impacts on the stated requirement. Each of these major decision-making areas is supported by specific guidance which will assist the architecture team in correctly evaluating the scope of the architecture. The next step determines how the architecture relates to the architecture and engineering scope and focus as shown below in Figure 2.2-2.

DoDAF V2.0 describes three 'tiers' that organize architecture development and governance in the Department (i.e., Department-level, Joint Capability Areas, or Component-level). These tiers, described in Section 1.3.1 and Section 4 of Volume 1, are central to the Federated approach[2] the Department has determined as the means of ensuring integration of effort in architecture development. Short definitions of the tiers are shown to the right in **Figure 2.2-1**.

Department-level Architecture is a type of architecture that describes processes applicable to the Department and Joint Staff as a whole. These architectures include the Global Information grid Architecture (GIG), the DoD Information Enterprise Architecture (DoD IEA).

Capability/Segment Architectures are those types of architectures that define and describe specific capabilities required by the Department for business, procurement, and tactical operations. The capability architecture is considered a segment architecture, as defined in OMB Circular A-130.

Component Architectures are those types of architectures that describe and define the business and operational functions of the Major components of DoD (i.e., Principal Assistants, Military Services and Joint Commands, and agencies of the DoD.

Figure 2.2-1: Architecture Tier Definitions

At each tier of the DoD, goals and objectives, along with corresponding issues that may exist, are addressed in establishing the scope and purpose of an architecture, as shown in the notional diagram in **Figure 2.2-2**.

One additional concern in establishing scope involves the development of an architecture that is intended to be a part of a larger architecture. When a requirement is defined, it may happen that the requirement is large enough that it must be completed in stages or phases, each of which may require an architecture. Determining the scope of an architecture in this type of scenario means that the architect and team need to be concerned not only with the larger picture involving the tier relationships, but also that each architecture created as a part of the execution of the requirement has linkages to the other architectures also being created. Since an architecture consists of data and information, use of that data and information consistently across architecture efforts will ensure that the requirements can be successfully executed.

[2] GIG Architecture Federation Strategy, 1 August 2007.

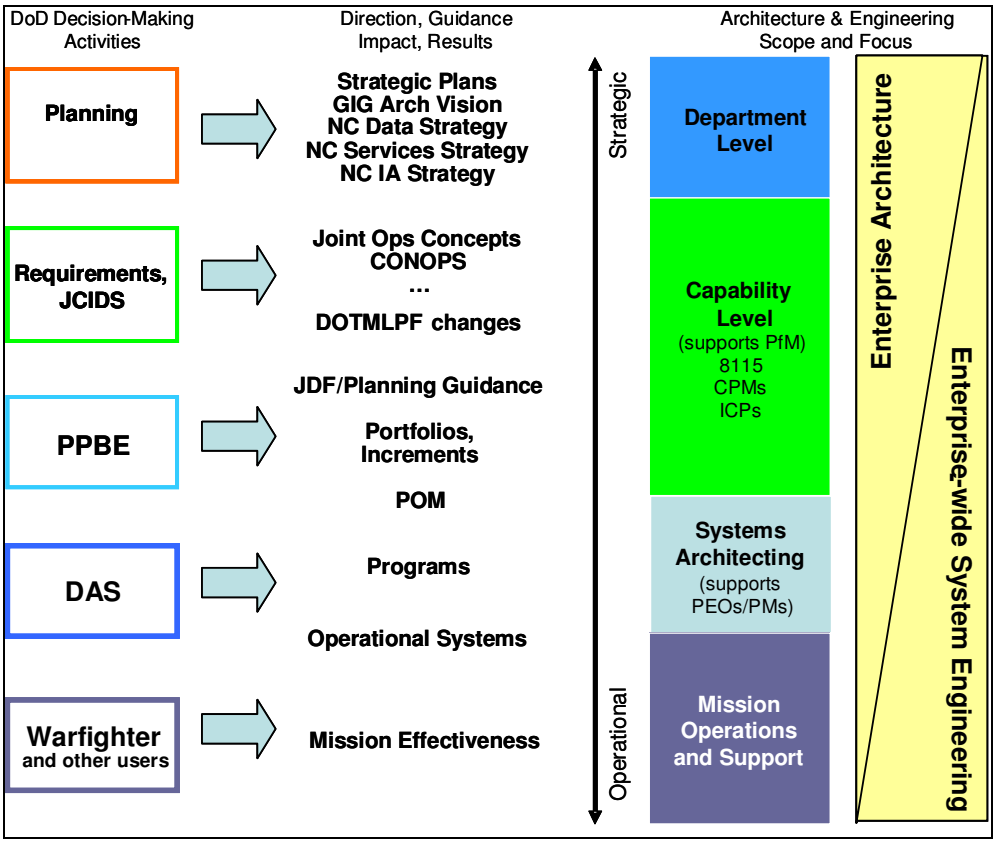

Figure 2.2-2: Establishing the Scope for Architecture Development

The scope of an architecture effort at any tier includes:

- Demonstrating how to achieve the intended use of the architecture.
- Definition of the boundaries (Purpose and Use expected) of the architecture.
- Data categories needed for analysis and management decision-making.
- Key players whose input, advice, and consensus is needed to successfully architect and implement change (i.e., Stakeholders, both internal and external).
- Goals and objectives of the effort, consistent with both boundaries and stakeholders.
- Level of complexity for data collection and information presentation needed for accurate analysis and decision-making.
- Understanding of external requirements that may influence architecture creation.

An architecture developed for an internal agency purpose must still be consistent with and mappable to the DoD Enterprise Architecture (DoD EA – e.g., Departmental-level architectures, Component Architectures, and Joint Capability Architectures). It must also meet any external requirements, such as upward reporting and submission of architectural data and models for program review, funding approval, or budget review due to the sensitivity or dollar value of the proposed solution. Volume 2 of DoDAF V2.0 contains guidance on data collection for specific views required by instruction, regulation, or other regulatory guidance (i.e., JCIDS, Exhibit 43/53 or Exhibit 300 submission; interoperability requirements, etc.).

An important early decision by the process owner and architecture team is the type of architecture being created to meet the established requirement. There are a number of types of architectures, such as those shown in **Figure 2.2-3** at the right. These are not 'Formal' types of architectures as presented in DoDAF, Volume 1, but serve as examples of commonly described names for architectures. These architectures can exist at any tier of the Department.

As we have already indicated, there are also levels of architecture that define at what level the architecture is being created, and how it will link to other tiers when completed.

Operational Architectures – Those architectures that describe the major processes and activities that drive business/tactical operations

Systems Architectures – Those architectures that describe the systems, networks, and other communications capabilities that support operations

Services Architectures – Those architectures that describe internal and external services capabilities that facilitate processes

Solution Architectures – Those architectures that define and describe future desired changes that impact on future business/tactical operations that are different from current baseline operations

Figure 2.2-3: Types of Architectures

Type and level of architecture determine which architecture you will create. Also, remember that an architecture is composed of data and information that will be useful to management analysis and decision-making. Models, views and documents represent the collected data for that purpose.

As an example, you are an architect leading a team to develop an architecture supporting process changes to Military Real Property Asset Management, one segment of the Logistics Joint Capability Area (JCA). You need to take several steps to ensure that you are creating the correct architecture views. These are:

- Determine WHAT TYPE of architecture should be created. Here, it appears that an Operational Architecture is appropriate because the emphasis is on process change. CAUTION: Determine if and how services play in the process improvement and be prepared to develop service views as required.
- Determine WHAT LEVEL of architecture is required. Here, it appears that you will be developing an architecture that which relates to the Capability Architecture Tier, and you will need to seek out appropriate guidance from Joint, DoD, and Military Service publications and guidance at several levels of the Department to assist you in your task.
- Evaluate whether or not the architecture can be done as one project, or needs to be broken into separate projects that will later be integrated. If the architecture is to be broken down to several efforts, then a high-level capstone architecture construct needs to be created of the top-level processes to which each architecture effort can be linked as created.

- Identify EXISTING ARCHITECTURAL DATA AND MODELS or views that already exist that may impact on your architecture, or require you to integrate your effort through existing data. Here, both the logistics community of interest (COI), and the business operations COI have major efforts and reporting requirements that involve standard data sets. Identification of these early prevents unnecessary work in developing unique data that later have to be changed to align to other architectures and reporting requirements.
- Begin creation of the All-Views (AV)-2 Integrated Data Dictionary to capture data definitions on existing data from previous architecture efforts and COIs. Consider the AV-2 as the organized sum of all architectural data collected in the development effort. (e.g., Process/activity data from OV-5 views, networking data from SV-1/SV-2, etc.)
- Create an All Views (AV) -1 – Overview and Summary Information that can be used to register the architecture effort with the Defense Architecture Registry System (DARS).

Architecture projects should be scoped to facilitate and support decision-making, and ultimately mission outcomes and objectives (**Figure 2.2-4**). Architectural data and supporting visualization models created from organizing data into useful information should enable domain experts, program managers, and decision makers to utilize these architectures to locate, identify, and resolve definitions, properties, facts, constraints, inferences, and issues both within and across architectural boundaries. This includes existing data that may be redundant, conflicting, missing, and/or obsolete.

Analysis is designed to uncover the result and impact of change ("what if") when something is redefined, redeployed, deleted, moved, delayed, accelerated, or no longer funded. Having a disciplined process for architecture development in support of analytics will produce quality results, not prone to misinterpretations, and of high value to decision makers in describing mission outcomes.

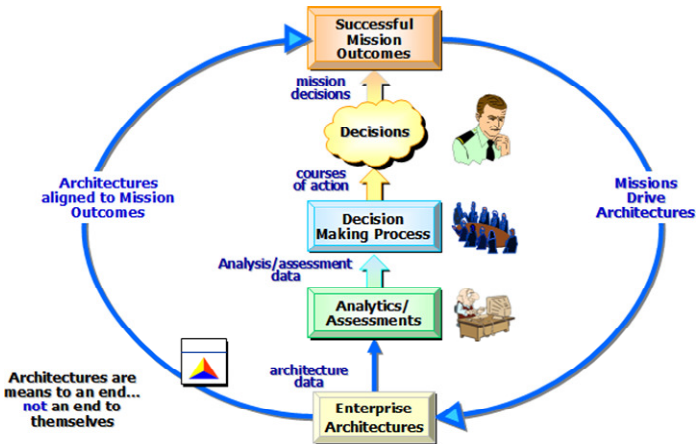

Figure 2.2-4: Mission Outcomes Supported by Architectures

"Fit for Purpose", then, means that the work you do to create an architecture by collecting pertinent data results in an answer for the requirement, concern or problem. It may not, on analysis, be the desired answer, but it should be a correct answer—one on which management can then make decisions on next steps.

The checklist shown in **Figure 2.2-5** will help you to complete Step 2.

Checklist for Step 2: Scope

___At what level is the architecture being developed?
- ___Department
- ___Component
- ___Joint Capability

___Is this architecture:
- ___Enterprise Architecture? ___ New ___ Update
- ___A Solution Architecture? ___ New ___ Update
- ___A Segment Architecture? ___ New ___ Update
- ___Part of a large architecture under the same requirement?

___Determine type of Architecture (operational, System, Service, etc?
___Who/What are your sources for developing the Architecture?
- ___Subject-matter Experts ___ Data Sources
- ___Other Architectures ___ Documents, Doctrine, Etc.

___Create AV-1, overview & Summary Information and register effort
___Create AV-2, Integrated Data Dictionary Template for collecting architecture data.

Figure 2.2-5: Checklist for Step 2: Scope

2.3 Determining Required Data (Step 3)

During the Scoping step of the methodology, potential existing architecture and COI data was identified for possible relevance to the architecture effort. The scoping effort took the stated requirement(s) and translated them into a more useful understanding of what the project entailed, and how to proceed. With that information now identified, the architecture effort registered (AV-1), and a template for an AV-2 (Integrated Data Dictionary) available, it is time in this step to clearly identify what data is required, where existing data is located, and how the definition of that data will satisfy project requirements.

Step 3: Determine Required Data to Support Architecture Development

The process of data identification and collection starts with simple steps, and gradually becomes more complex. The first step is to determine what categories of data are needed for the architecture. This is not a technical exercise, but rather a process that organizes potential data into 'buckets' (e.g., medical data, housing data, logistics data, payroll data) that serve as a beginning point for later organization and definition of individual data needed. The DoDAF Meta-model (DM2) provides a set of data categories that is useful in

developing this approach to data identification. (e.g., Performers, Activities, Services, Capabilities, etc. as found in DoDAF Volume 2.)

For an existing process, initial collection of data may be nothing more than identification of data already contained in documents, databases, or other sources routinely used to execute a process or activity. In a new process, the initial selection of data categories may be more of a calculated guess, but as the process of selecting and refining data occurs, the actual data needed will become readily apparent.

Data is organized through development of an Integrated Data Dictionary (DoDAF calls this an AV-2) which can be as simple as a spreadsheet, or be completely automatically in a toolset that organizes the data into an AV-2 format. **Figure 2.3-1** below shows a simple AV-2 format.

> NOTE: DoDAF V2.0 does not prescribe any particular toolset for use in developing architectural views and architectural descriptions. There are a number of commercial toolsets on the market, many of which have the capability to create and automatically maintain a data dictionary through their internal databases.

When individual data categories are initially identified, a spreadsheet is often a simple tool that can quickly organize the categories for later use. Most spreadsheet applications provide multiple 'pages' that can be labeled as categories and formatted as shown in Figure 2.3-1. The headings needed are:

- Data Name.
- Data Definition.
- Data Source.
- Data Attributes (The characteristics of data, such as Alpha-numeric, number, Boolean (Y or N), etc.).
- Data Metric (How the data is expected to be used, or the result expected of it).

At first, the data categories and some existing data being used in existing processes and some sources may be all that is quickly discovered by the architecture team. Creating the AV-2 template **to use in initial identification of data categories** will help in ensuring that data is completely identified for use later in developing models and views of the architectural data.

AV-2 Integrated Data Dictionary					
Data Name	Data Definition	Data Source	Data Attributes	Data Metrics	Characterization ("Input", "Output")

Figure 2.3-1: Notional AV-2 Integrated Data Dictionary

A second concern is the level of detail of data required for use during the effort. The level of detail will be determined consistent with the level of effort for the architecture itself. If the architecture is intended to be a 'high-level' view of a process, and will not have subordinate activities broken down to individual work actions (called decomposition), then data need not be similarly decomposed into more specific data elements. As an example, **Figure 2.3-2** shows both a decomposed process/activity and an accompanying data element that might be required.

Process Activity	Data Required
Process Accounts Receivable	List of Accounts
	List of Outstanding Invoices
	Customer Name
	Amount Received
	Check Number
	Account Number
Receive Check	Invoice Number
	Customer Name
	Amount Received
	Check Number
	Account Number
Process Check	Invoice Number
	Account Number
	Invoice Number
Cross-reference to billings	Amount Received
	Account Number
	Invoice Number
Certify Receipt	Amount Received
	Customer Name
	Amount Received
	Check Number
	Account Number
Submit Transaction	Invoice Number

Figure 2.3-2: Decomposing Process/Activity and Data Elements

Collecting data is a lot easier with an existing process rather than a new one. In the next section (Section 2.4), we will look at both new data collection, and assembling data from existing processes. As we have already seen above, we collect the same information on data regardless of which type of requirement exists.

2.4 The DoDAF Meta-model

An aid to defining and collecting data consistent with DoDAF V2.0 is proved by the DoDAF Meta-model (DM2). This meta-model (a model about data), replaces the Core Architectural Data Model (CADM), a storage format, referenced in previous versions of DoDAF. DM2 is a replacement for the CADM, but does not provide a physical data model. Instead, a Physical Exchange Specification (PES) is provided as an exchange mechanism, leaving the task of creation of a physical data model to the tool vendors. DM2 provides a high-level view of the data normally collected, organized, and maintained in an architecture effort. It also serves as a roadmap for the reuse of data under the federated approach to architecture development and management. Reuse of data among communities of interest provides a way for managers at any level or area of the Department to understand what has been done by others, and also what information is already available for use in architecture development and management decision-making efforts. Finally, the DM2 can be used to ensure that naming conventions for needed data are consistent across the architecture by adoption of DM 2 terms and definitions.

The DoDAF Meta-model has three views:

A ***Conceptual Data Model (CDM***) is described in Volume 1, and defines the high-level data constructs from which architectures are created, so that executives and managers at all levels can understand the data basis of architecture. The CDM defines concepts and describes their relationships in relatively non-technically and easily understood terms.

A ***Logical Data Model (LDM)*** adds technical information, such as attributes to the CDM and, when necessary, clarifies relationships into an unambiguous usage definition. The Logical data Model is described in depth in DoDAF, Volume 2.

A ***Physical Exchange Specification(PES)*** is described in DoDAF, Volume 3, and consists of the Logical Data Model with general data types specified and implementation attributes (e.g., source, date) added, and then generated as a set of XSD's, one schema per model/view.

When starting an architecture development effort, and particularly if it is a first effort, take the time to understand the conceptual model and how data relates to other data. As you begin to plan the architecture effort, these categories of data (i.e., performers, activities, systems, services, data and information, etc.) described in DoDAF Volume 2, become the building blocks for your architecture.

2.4.1 Determining Data Requirements

Step 3 of the DoDAF Methodology discusses how to determine what data is needed to support architecture development. The DoDAF meta-model is a rich source for identifying and organizing that data for use in the architecture effort. What data you will need is determined by the kind of architecture being created, its purpose, and the expected results.

Much of the information you need has already been collected using the checklists for the first two steps. At this point, you should already know:

- The **name** of the architecture.
- The **purpose** of the architecture (Enterprise, Segment, Solution, or some specific architecture (i.e., systems, services, etc.))
- The **scope** of the architecture.
- The **level of architecture** development (Department, JCA, Component).
- The **communities of interest** (i.e., stakeholders) impacted by the development.
- **Target objectives** aligning architecture development with agency requirements.
- **Specific architecture requirements** (i.e., JCIDS, Interoperability, ISP, etc.) that must be accommodated in the development process.

Now, you need to add to that list. Two different actions come into play here. First, the data needed to define the proposed architecture needs to be identified and described. That data may be activity data, system or service data, network data, or any of a number of other data categories that contain appropriate data. Second, you are going to want to collect data as well that will enable visualization of collected data for decision-making purposes. Collection of this data occurs through use of the All-Views-2, The Integrated Data Dictionary that is described in Volume 2 of DoDAF.

As you can see in the DoDAF Conceptual Data Model below in **Figure 2.4.1-1**, these categories sometimes overlap. We will discuss visualization of data later in Section 8, Documenting the Architecture. For now, we want to limit ourselves to finding out where we get the data we need.

Within the CDM, there are a number of categories of data that interact with each other to produce an architecture. Most architectures require a number of these categories to completely describe the purpose and use of the architecture, and be able to use the architecture for management analysis and decision-making.

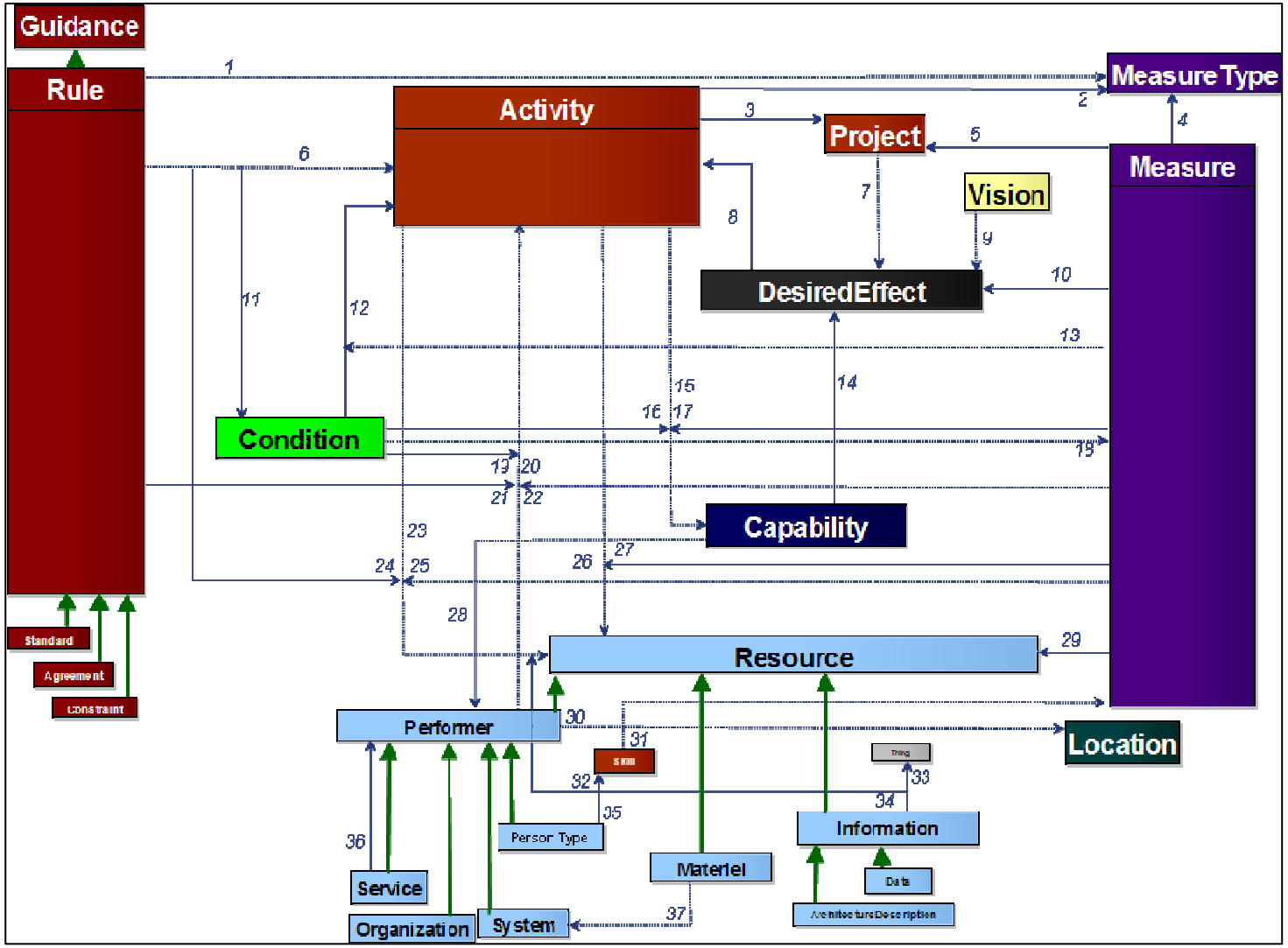

Figure 2.4.1-1: DoDAF Conceptual Data Model

Looking at the CDM, it becomes immediately obvious what will impact your architecture. All architectures, regardless of type, use performers to execute the processes, networks, or service requirements generally contained in an architecture. Architectures also define rules that specify how work is performed, the rules that describe how activities are performed, and the locations in which activity occurs, whatever the type.

The conceptual model also provides a way to capture information on the architecture project itself. Volume 2 of DoDAF takes these categories of data, organizes them according to how the data will be used, and how to document that data for later reuse or analysis.

Let's look at some examples of how this occurs

Example 1: Suppose you are creating a *solution architecture*—an architecture that describes how a particular project with a specified set of expected results is constructed to facilitate the build of a solution to some problem or specific requirement. Data categories could include: Project, Goals, Objectives, and Measures.

Example 2: Suppose you are creating a *segment architecture*—an architecture that describes a broad program or group of activities that relate to a single business area of an organization. Data categories could include: Capabilities, Activities, and Organization.

Example 3: In this architecture effort, a systems architecture is being created to document how a system or group of systems provides enhanced functionality to a business or operational practice. Data categories could include: Performers, Systems, and Activities.

Now let's go to the checklist (**Figure 2.4.1-2**). Checklist 3, below, collects information on the kinds of data you will need for your architecture project. It is organized in sections; you need to complete only those sections that apply to your project.

Checklist 3: Identifying Data Required for an Architecture

Architecture name: (Checklist #1)
Purpose: (Checklist #1)
Scope: (Checklist #2)
Level of architecture: (Checklist #2)
Communities of Interest: (Stakeholders – Checklist #1)
Target objectives: (Checklist #1)
Specific Architecture Requirements: (Checklist #1)
___Determine data categories that impact the architecture:
_____
___Select appropriate metadata items
___Define needed entities
___Determine needed level of detail
Establish high-level measures/metrics for architecture

Checklist 2.4.1-2: Identifying Data Required for an Architecture

The majority of the information needed is from information contained in the previous checklists. Determining the data categories requires access to the DoDAF Volume 2, Section 2, Meta-model Data Groups as a guide. In that section, there are 12 data groups:

- Performers
- Resource Flows
- Information and Data
- Training, Skill, Education
- Capability
- Services
- Project
- Goals
- Rules
- Measures
- Locations
- Activities

These data groups are the first level of organization for data needed within an architecture. Volume II, Section 2, provides a complete description of the data group, its use, and its data capture method. These data groups serve three distinct purposes:

- Collect data about the architecture itself (i.e., project, goals, and measures).
- Collect data about the process, system, service, or other requirement that utilizes the architecture.

- Describe how data groups interact (i.e., relate or associate) with each other to transform individual pieces of data into meaningful information for analysis and decision-making.

The data groups also describe *meta-data*, which is, in simple terms, data about data. These are high level terms and description that are intended to serve as examples and templates for creation of data within a specific architecture.

As an example, a PERFORMER is a central block in the building of an architecture. It is the WHO in the development process. ACTIVITIES are assigned to performers to accomplish some desired outcome. In turn, Activities may be supported by SYSTEMS or SERVICES. These are the WHAT that produces the desired outcome or output. Activities are performed by performers utilizing RULES to describe the steps taken to perform some action, and perhaps MEASURES designed to evaluate the effect or correctness of the outcome or output. Performers execute an activity at some LOCATION. The CDM and the information contained in Volume 2, Section 2, provide the means for determining what data among the data types will be needed for a specific architecture.

The next step is to determine specific data needed, based on the groups. If you need a PERFORMER to execute an action, WHAT KIND of Performer is needed? Is there more than one, or is there a single Performer, but at several levels? A performer can be a person, a system, an application, a service, or another thing that acts to execution a task or action.

The final part of this step is to create some high-level measures of performance. What is the level of expectation or result that should occur when this action is executed?

When creating an architecture, it is common to start with the process that most impacts on the requirement, along with those other processes that directly support the principal process. Then the standard interrogatory questions are applied:

- WHO executes the activity (Performers, Systems, Services)
- WHAT is it that is required? (Activities)
- WHEN is the data required? (Projects)
- WHERE is the activity executed? (Location)
- WHY is the data required? (Capability, Goals)
- HOW is the activity executed? (Rules)

2.5 Collecting Architectural Data (Step 4)

During the scoping of the architecture, described in Step 2, key stakeholders were identified who impact on the actions described in the architecture under development. These stakeholders include both internal and external sources, and these sources are the primary source of needed data. They become the ‘subject-matter experts’—those whose knowledge of the processes or activities are the most current and accurate. They possess the knowledge on how the process works, what is required to execute it, and what data is needed to execute the actions. Importantly, they may also have specific requirements for

models and views in a given development project (i.e., JCIDS Gatekeepers, or BTA requirements).

Collecting data, then, becomes primarily a matter of clearly understanding how stakeholders need or use data, and how that data is later presented to them in a form they can understand and utilize in business decision-making. In this section, we discuss the collection of data, both data for existing processes/systems/services, and architecture development for new processes/systems/services.

Two things drive architecture—data that defines the architecture and definitions of the data that are understood and accepted by others in their own architecture development.

Step 4: Collect, Organize, Correlate, and Store Architectural Data

The first step in collecting architectural data is to determine how stakeholders (i.e., subject-matter experts) collect and use data in their normal activities, or receive it from a primary source for their specific purpose. In the case of a new process, identification of prospective data and how stakeholders expect to use the data serves the same purpose.

Not all users of data are going to use that data in the same way, or for the same purpose. Some may use data originally created for another purpose for their own requirements, since their understanding of the data is consistent with the creator and useful for another purpose. In order to ensure that all the data needed for the primary purpose of the architecture, both the direct stakeholders, and those who use portions of the data for other purposes are consulted on the data collection requirements. The tables shown in **Figures 2.5-1** through **2.5-2** below show the steps generally involved in data collection, regardless of the status of the architecture (i.e., existing or proposed).

Checklist for Step 4, Collecting Architecture Data

___Review Process (Either existing process undergoing change, or proposed new process steps for data requirements
___Establish data categories to use for data collection effort (Vol 2 DoDAF)
___Search DARS and DMR for existing or similar architecture efforts and associated data
___Initiate AV-2 structure for collecting architecture data entities, associations and relationships (Vol 2 DoDAF)
___Determine Data instance resources required to support process, system, services

See individual checklists below for existing and new process data requirements

Figure 2.5-1: Checklist 5 Collecting Architectural Data

2.5.1 Collecting Data on Existing Processes

Figure 2.5.1-1 provides the steps for collecting data on existing processes.

Checklist for Step 4 – Collect Data for Existing Processes

___Review documentation of existing process
___Identify data used in process steps using AV-2 template
___Identify external data, COI data, and other sources for data used by the process, or needed by the process
___Document differences (if any) in definition or structure
___Validate data list with Subject-matter Expert (SME)

Figure 2.5.1-1: Data Collection Checklist for Existing Processes

Documentation is always the best source for information on data required to execute a process or activity. Existing architectures created under a previous version of DoDAF or another architecture framework will usually contain a data dictionary somewhat close to the notional one in Figure 2.5.1-2. Toolset vendors generally have similar capabilities to generate a spreadsheet with data and definitions for models created in their tools. If these documents are not available, then a procedural guide, process descriptions, forms used, or other documents that describe activity and data can also be useful in identifying data.

Name, DM2 Concept	Description
Administer Process Mgt Program (PWS 6.1.2.3) OPERATIONAL ACTIVITY	Collaboration, communication, and coordination. - Clearly communicate AEDC and contract priorities. - Ensure all contracted staff actively collaborate, communicate, and coordinate with each other and their Government counterparts. Ensure information exchanges are timely, thorough and accurate.
Advanced Planning & Scheduling SYSTEM FUNCTION	Identify and prioritize all maintenance, repair, and performance improvements needed to meet current and projected operational needs for RDT&E and Test Support assets. Identify the impacts of requirements not funded and recommend alternatives to satisfy those requirements, including incremental solutions that may be feasible. Alternatives shall include critical timelines for accomplishment. [REF: ATA Procedure, P02-6001, 20 Dec 2004]
Advocate Center/Laboratory requirements at AFMC OPERATIONAL ACTIVITY	The Deputy CIO will advocate Center/Laboratory requirements at AFMC in the pursuit of Command enterprise solutions as opposed to the propagation of non-standard local development efforts.
Aeropropulsion O&M PERFORMER	Test operations and maintenance of assets used in direct support of TE20. Safely provide sustainment of test facility operations and maintenance, ID&C systems operations, test article and test unique facility build-up and installations, procedures writing, and hazard analyses, while providing performance workforce to move within TE to support all product areas.
Aeropropulsion Plant Ops Br PERFORMER	Responsible for plant support of all Aeropropulsion Products Branch testing as well as vacuum plant support for J-4 and J-6 of the Space and Missiles Products Branch. Provide single point integration for operations and maintenance of the ETF and ASTF plants.
Aeropropulsion Products Branch PERFORMER	Management of the Aeropropulsion T&E Branch

Figure 2.5.1-2: Notional Data Dictionary Example

Data collection is generally started based on the EXISTING process so that the full listing of data becomes known, and can be documented in the AV-2, or as part of the data collected for other views. When completed, this will represent the baseline ("As-Is") view of the architecture. Data collected here can be internal data (i.e., created and managed directly by the process owner), external data (Data owned by another process owner, but used within the process being reviewed), COI data (Data registered with a specific COI, and expected to be used by all architecture developments that impact the

COI), or foreign/coalition data (Data that is not owned by the Government, owned by a foreign government, or a non-governmental organization (NGO) that may impact on the architecture.

As this data is entered into the AV-2, its source and definition should be entered in the appropriate blocks. Where data is used from several sources, it is important to review that the definition of the data provided by the process owner is the same as or equivalent to the definition used in the process. Differences should be entered in definition and source columns.

This is also the time to enter the *data attributes* and *data metrics* in the AV-2. Most data dictionaries define the format (i.e., alpha-numeric, numeric, Boolean (Y or N), etc. for the data item. These attributes will be important in development of any systems or services that may be required as an output of the process improvement that follows architecture development. Also important are the data metrics, the expected outputs of the data item. Metrics provide the means for assessing both the pertinence and quality level of data required in any given process or action. Metrics also determine if the expected data is actually the output and available for other needed actions.

Once everything has been collected, then it is time to meet with the subject-matter experts (SMEs) to review and validate the data collection. This is especially the case where data collection involved a review of documentation rather than import of data from data dictionaries. SMEs can tell whether or not the data is really used, and also often know other, 'unofficial' data sources that may be used in routine process execution.

2.5.2 Collecting Data for New Process Development

Figure 2.5.2-1 provides the steps for identifying and organizing data when an architecture is being created for a new process. A complete description of each step follows the checklist.

Checklist for Step 4 – Collect Data for new Process Development

___Review requirement statement to ensure understanding
___Develop a set of process steps needed for process execution
___Identify process inputs, outputs, controls and resources
___Identify data used in process steps using AV-2 model
___Identify external data, COI data, and other sources for data needed by the process
___Document differences (if any) in definition or structure
___Validate data list with Subject-matter Expert (SME)

Figure 2.5-2-1: Data Collection Checklist for New Process Development

Development of a new process starts with the requirements statement or problem. That requirement statement must be clear enough for the development team to understand what is required, and then to fashion a set of process steps that might occur during

execution. Often, this means taking a similar process and looking at the steps used in that process for execution, and then adjusting, extending, or otherwise changing those steps to meet the objectives of the requirement.

Development of new processes is one of those actions that will provide or disprove the accuracy, clarity, and intent of the requirement statement. If the clarity of the requirement statement is insufficient, then the team needs to meet with the process owner, or the developer of the requirement statement, and clarify intent, get an updated requirement statement, and move forward. Usually during these discussions with the process owner or writer, it will become obvious what the major steps entail, and the team will then have to develop lower level activities to support execution of the process.

Describing the major process steps involves creating a simple time line, as shown in **Figure 2.5.2-2**, below. "Point A" initiates execution of the process, and "Point B" is the final step that yields the expected result. The challenge is two-fold; defining the steps needed to get from A to B, and how that set of actions occur. Think of the time line—the path of execution—as a rope strung between two stakes in the ground. Point 'A' defines the start of the development effort, includes scoping describing and registering the effort, and determining the expected result (which we call 'B'). In order to get from Point A to Point B, we have to know the steps to be taken, how these steps inter-relate, and how the final output or outcome is the expected result.

Movement from Point A to Point B takes one or more steps, each of which, since they are related, use their input and outputs to link the steps involved in the process.

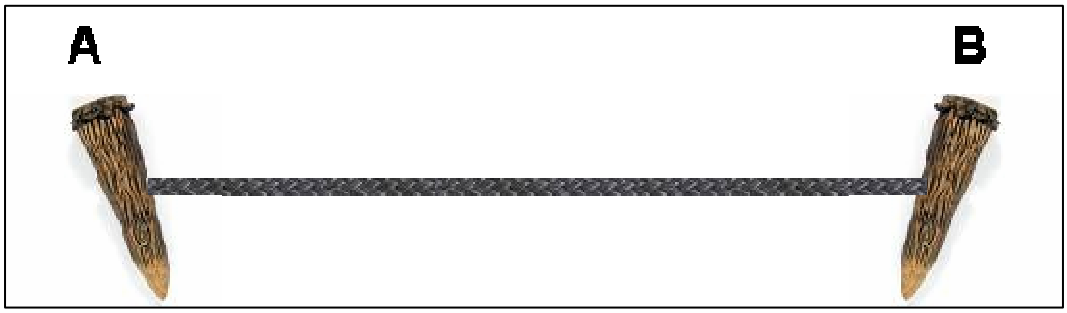

Figure 2.5.2-2: The Path of Process Execution

The description of a process involves data on several things. These are:

- The action(s) to be performed or executed (Activity).
- The performer of the action (Performer or Actor).
- The event that causes action to occur (Trigger).
- The data supplied to initiate the action (Input).
- The resources that are applied to facilitate execution of the action (Resource).
- The rules that apply in executing the action (Controls and/or Business Rules).
- The expected outcome or output of the execution of the action (Output/Outcome).

- The measures of effectiveness to be applied to determining if the execution of the action is/was successful (Measures).

This type of information can be collected in many ways, such as using existing data from Communities of Interest (COI) in the DoD Metadata Registry (DMR), reuse of data from similar processes used to define and describe the new process. To the extent possible, authoritative data sources should be used. These sources are 'recognized' sources of data that have already been validated for use. The DMR, the Uniform Joint Task Lists (UJTLs), and others described more fully in DoDAF can be used for this purpose. The subject-matter expert(s) in the process area under development are also a valuable source of appropriate data to use.

Data, when collected and described, is organized as a Data Dictionary, something DoDAF calls the All Views – 2, (AV-2), Integrated Data Dictionary. Use of the AV-2 Model Template for this purpose is recommended since it aligns with the DM2.

During and after data collection and documentation, validate the results with the subject-matter experts to ensure that appropriate data has been collected, and is properly used for the purpose intended in support of the requirement.

2.6 Conducting Analyses in Support of Architecture Objectives (Step 5)

The real value of architectural description development lies in the ability to use the data that forms the architectural description for program, management, and future planning analysis.

Step 5: Conduct Analyses in Support of Architecture Objectives

Architecture-based analytics is a process that uses architectural data to support decision-making through automated extraction of data from a structured dataset, such as the use of a query into a database. Well-designed architectural description which are consistent with the purpose for which they were created, are also well suited to the analysis process.

2.6.1 Types of Architectural Analysis

There are two categories of analytical activity. These are:

- Static Analyses: Those analyses, which are based on making a value judgment, based on data extracted from the architecture. For example, analysis of the weather patterns and measurements for the last 50 years to determine trends and correlations would be static analyses.
- Dynamic Analyses: Those analyses, which are based on "running" an executable version of the architecture to observe the overall behavior of the model. For example, the construction and execution of a dynamic weather prediction model to determine the possible future weather trends is an example of dynamic analysis.

2.6.2 Examples of Analytics

Analytics can be used in conjunction with many aspects of the architecting process. Examples of analytical support can be found within DOTLMPF, as shown in **Table 2.6.2-1**, below. DOTMLPF is the analysis of who (people, organization, leadership) perform what operations (doctrine) at which locations (facilities) using (training) which system resources (material) to produce and consume information and data. DOTLMPF analysis leads to better definitions of warfighting capabilities by being able to anticipate effects and assess impact of change on domains and by examining usage (who/what affects something) and references (who/what is affected by something). DOTLMPF domains map to DoDAF architecture elements with the following analytical support activities.

Table 2.6.2-1: DOTMLPF

DOTMLPF Domains	DoDAF Architecture Elements	Analytical Support Activities
Doctrine	Functions, Performers, Assets, Locations, Nodes	Examine Tactics, Techniques and Procedures
Organization	Performers, Org Units	Examine organizational structure
Training	Functions, Performers, Assets	Train personnel on their activities and the systems they use
Leadership	Org Units, Performers, Assets	Examine leadership issues
Materiel	Functions, Material, Data, Information, Location. Assets, Performers	Examine materiel solutions – a new system?
Personnel	Performers	Examine personnel solutions – new personnel or personnel with better qualifications
Facilities	Locations	Examine fixing, building or modifying facilities

It is not the intent for DoDAF to prescribe all possible analytical activities. The list above is only a partial listing of potential activities that relate to DoDAF architecture elements useful to the DOTMLPF Domains. As more demands are placed on architecture, and as industry spawns more automation, the flexibility described in DoDAF will encourage further innovation from architects and from tool vendors.

2.7 Architecture Analysis Overview

Step 5 of the DoDAF methodology, *Conduct analyses in support of architecture objectives*, defines several types of analyses that are useful, and which can utilize architectural data as an aid to decision-making. DoDAF Volume 1, Section 10 describes ***Architectural Analytics*** as the processes that transform architectural data into useful information in support of the decision making process. As shown below in **Figure 2.7-1**, the data collected through architecture development is the foundation for analytics.

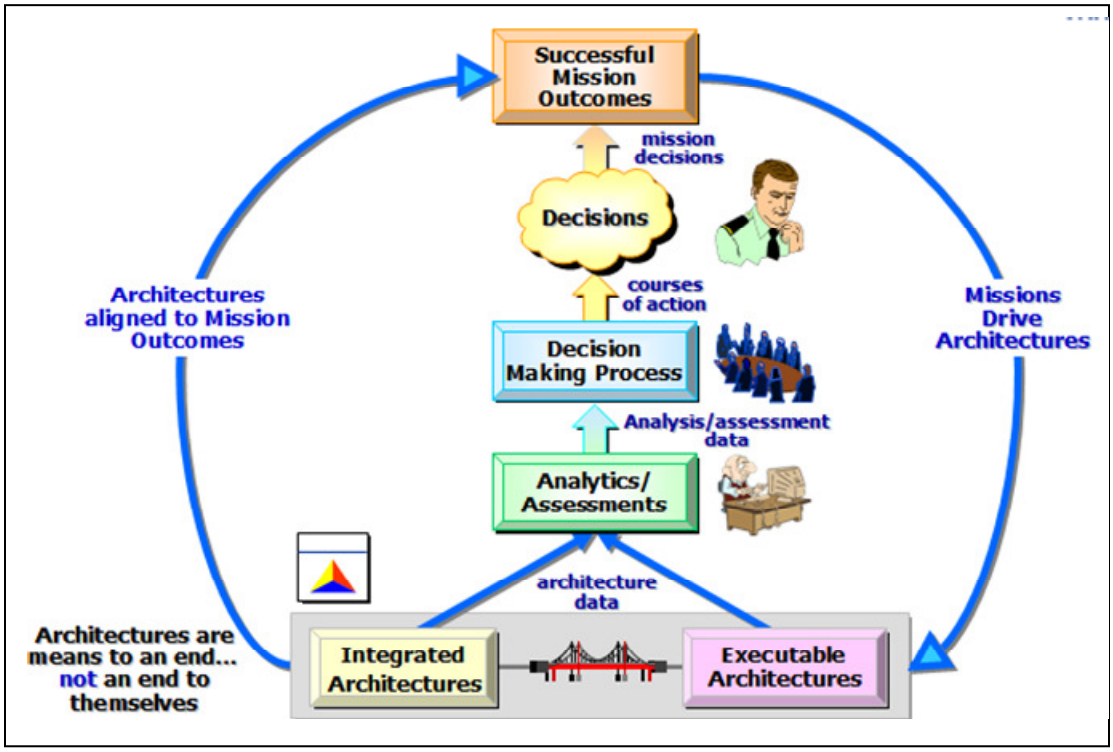

Figure 2.7-1: Analytics Process

There is a significant caveat to architectural analytics: ***Architecture development is an iterative process, evolved over time. Analyses developed from architectural data remain valid only as long as the processes and information do not change, and management decision-making remains focused on the same problem for which the architectural data was collected.*** When any of these variables (i.e., architecture purpose, process steps, information, or management direction) change, then previous analyses should be reviewed to determine if the previous analysis needs to be redone, based on the newly provided information. Constant feedback and examination needs to be understood as natural in an environment where program direction and priorities are constantly in flux.

Analytics can take many forms. Among the most common in the Department utilizing architectures are the Joint Capability Integration & Development System (JCIDs), Systems Engineering, the Defense Acquisition Process (DAS), and the Program Planning, Budgeting & Execution (PPBE) processes.

While each of these 'key' Departmental processes utilizes some of the same data from architectures, their analysis may be quite different according to the purpose for which the architecture was developed. Taking a simple example, **Table 2.7-1** shows the difference in data utilization. While the raw data collected among the key processes is often the same, its primary use is quite different.

Table 2.7-1: Departmental Process Data Collection and Use

Key Process	Collected Data	Primary Data Use
JCIDS	Capability Requirements Existing/Desired expertise Existing/Desired doctrine Existing/Desired training/ Education requirements	Capability Development
SE	Existing/Desired Systems capability data Systems Requirements Data	System Development
DAS	Acquisition Data	Procurement/Acquisition management
PPBE	Budgetary Data	Budget Submission/Execution and Oversight

The other example analyses described in the DoDAF methodology, such as:

- Shortfall Analysis
- Capacity Analysis
- Interoperability Analysis
- Business Process Analysis
- Test Architecture Completeness, Accuracy, and Sufficiency

can also use the DM2 data categories to organize data for the analysis. Use the descriptions of the various analyses in Volume 1, and the examples in the DoDAF Journal to determine the appropriate categories of data for each analysis. As in the example above, some data changes according to the analysis desired, but the basis structure of data needed is essentially the same.

2.8 Documenting Results (Step 6)

DoDAF, Volume 1, Section 8, describes various types of presentation formats for architectural data being transformed and utilized as architectural descriptions. Similarly, Volume 2, and the DoDAF Journal contain further descriptions of data, format, and their use in example architectural descriptions.

Step 6: Document the Results

The first steps in determining how architecture models will be populated as views or other documents is to look toward the desires of management and the common practice of the organizations for presentations is used in analysis and decision-making. Virtually every organization has a different approach, although most share in common characteristics. The Chief architect, in discussion with the process owner, will determine the approach for documentation of the architectural description.

Architectural descriptions all share some common characteristics. At a minimum, they are documents that:

- Utilize, organize, and prepare for sharing information collected on the project (Architectural data).
- Explain the purpose, objectives, and requirements of the project for which they were created.
- Present a common viewpoint throughout the Architectural Description as desired by the process owner.
- Are created as presentations of data that will be meaningful to managers and other non-technical reviewers for analysis and decision-making.

There are any number of ways to present data, consistent with agency preferences and the need for complete understanding of the concepts underlying the architecture under development. These include:

- Text documents.
- Spreadsheets.
- Dashboards.
- Composite Diagrams.
- Fusion Diagrams.
- Models.

All of these presentations types utilize data collected during architecture development effort for their underlying factual basis. Fusion and composite diagrams may also use data not contained in the architectural data collection, but pertinent for analysis and decision-making purposes. (e.g., an agency may be redesigning a process that is similar to one in another agency, and wants to compare their process steps, performance measures or other characteristics with those being designed in the present project.)

DoDAF V2.0, Volume 2 describes over 50 examples of varying models that can be used in developing architectural descriptions. These examples include all of the 'products' of previous versions of DoDAF, along with new models and views that can be created that relate to similar models and views from MODAF, NAF, TOGAF, and other frameworks.

3. ARCHITECTURE PLANNING

The key to a successful architecture development effort lies in effective planning BEFORE any work is done. Architecture planning is not a separate step within the DoDAF methodology. Rather, it that series of steps that must be taken to ensure that the organization, administration, and operation of the architecture development process can achieve success. Architecture planning includes a number of actions, such as:

- Selection of a team.
- Meeting with the process owner to determine the scope of the effort (DoDAF Volume 1, Section 2).
- Determining the project purpose, goals, objectives, and guiding principles (DoDAF, Volume 1, Section 3).
- Defining the type of analysis needed or desired from the effort in order to make meaningful decisions for action (DoDAF, Volume 1, Section 10).
- Creating a work breakdown schedule to determine the time, resources, and other boundaries of the effort.
- Adopting a methodology, techniques, and other resources that will aid the team in delivering the desired data, models, and presentations to support management decision-making (DoDAF, Volume 1, Sections 3 & 7).
- Conducting training on DoDAF and tools/techniques that will be utilized to produce DoDAF-conformant data.

Each of these actions is a key contributor to the overall effort. All architecture efforts start with the designation of a ***process owner***—that manager or executive who is directly responsible for the operation of a process. It is generally the process owner that determines the need for an architecture effort, designates the team leader (and often the team itself), an architect, and purpose goals and objectives of the effort.

Checklist for Architecture Planning

___Select a team leader, team, and chief architect
___Meet with the process owner to determine the scope of the effort (DoDAF Volume 1, Section 2)
___Determine the project goals, objectives, and guiding principles (DoDAF, Volume 1, Section 3)
___Define the type of analysis needed or desired from the effort in order to make meaningful decisions for action (DoDAF, Volume 1, Section 10)
___Create a work breakdown schedule to determine the time, resources, and other boundaries of the effort
___Adopt a methodology, techniques, and other resources that will aid the team in delivering the desired data, models, and presentations to support management decision-making (DoDAF, Volume 1, Sections 3 & 7)
___Conduct training on DoDAF and tools/techniques that will be utilized to produce DoDAF-conformant data

Figure 3.1-1: Checklist for Architecture Planning

4. ADDRESSING RISK THROUGH ARCHITECTURE

4.1 What is Risk Management?

"Risk management is a discipline for living with the possibility that future events may cause adverse events."[3] Avoiding those kinds of problems can be enhanced by using architectures in concert with a *Risk Management Program.* Whenever change is considered, it is important to consider, evaluate, and manage any risks that may be associated with that change. ***Risk Management*** is the act or practice that deals with those risks. In the architectural community, ***architectural risk management*** is the process that normally occurs concurrently with architecture development, which, through analysis, identifies flaws in the architecture that may have significant impact on the execution of the improved process, system, or service.

There are specific steps associated with risk management programs that are discussed briefly below, and with more depth in the DoDAF Journal. These include:

- Risk Management Planning.
- Risk Identification.
- Risk Assessment.
- Risk Quantification.
- Risk Response Planning.
- Risk Monitoring & Control.

As risk management is discussed below, keep in mind that it is seldom possible, except in the most extraordinary situations, to completely eliminate risk. In fact, it is more usually the case that elimination of all risk is too costly and burdensome for most projects. Instead, a good risk management program, as described below, identifies way to reduce risk, manage others, and establish responsibilities for ensuring that identified risk does not get out of control. Thus, addressing risk generally involves one of three approaches, ***acceptance, mitigation, or transfer***, each of which is discussed in conjunction with the steps above

- ***Risk acceptance*** means that the risk is understood, and the process owner/developer is willing to proceed with the original plan despite the risk. In this case, risk is continually assessed to ensure that the level or intensity of risk does not change in a way that would materially affect the desired process change or development.
- ***Risk mitigation*** involves developing ways to reduce risk by making change in the target process and/or development plans. Once the risk is clearly understood through analysis, then specific steps are undertaken to reduce or eliminate the risk. That may involve changes in the architecture, the transition plans that define the steps necessary to execute change, or a plan to continually monitor change as it occurs to ensure that execution follows exactly the desired solution description. Like acceptance, risk mitigation requires firm, consistent oversight and continual review.

[3] Turk, W. (2006). Risky Business. *Defense Acquisition & Technology*. Office of the Under Secretary of Defense (Acquisition, Technology, & Logistics) (December 2006). Pp. 32-35.

- ***Risk transfer*** is the identification of specific risks through analysis, and the assignment of those risks to others outside the process. Risk transfer often is identified in architecture development as a break or gap—a lack of useful data, or validated data that comes from an external source. The solution to the problem is to identify the risk to the external source and ensure that any data used follows the established conventions of the architecture and the process change. In this case, risk is periodically assessed from the perspective of data review.

4.2 Risk Management Planning

A Risk management Plan is critical to the success of any EA effort. The plan defines the project, stakeholders, expected or desired results, objectives of the project, the organizational structure of the team and supporting elements, and roles and responsibilities of all participants, to include the executive sponsor or process owner. The plan, when complete, will also discuss the level of risk acceptance that can be accommodated during the effort, steps to be taken to mitigate risk, and potential transfer actions that will reduce risk.

4.2.1 Risk Identification. Identification of potential risks early in the project can prevent later failure. Risks include those resulting from the execution of actions described in the architecture, potential effects to others resulting from the proposed change, or even internal risks, such as the ability of the team to influence change, or identification of key personnel whose support is required for change. Risk identification must include information of potential managers of risk internally, as well as possible people and organizations that can assume elements of risk identified during planning.

4.2.2 Risk Assessment. This step evaluates identified risks in terms of the *probability* that risk will exist in the project, and the *impact* that a risk poses to the success of the program or project. Many project risk assessments utilize a simple 'x-y' axis matrix to determine the extent and importance of risk. Defining both probability and impact in levels ranging from 'low' to 'high' provides a means to determine what action may be required. A combination of probability and impact (which together define severity) will determine whether the risk can be ignored, or will require close monitoring, either within the project, or by an outside group which accepts the risk.

4.2.3 Risk Quantification. Risk Quantification takes the initial assessment matrices and analyzes potential effect. In architecture planning, one of the most common concerns is the 'domino effect' (i.e., the effect actions taken have on other activities) and how to reduce the turbulence and undesired results that may occur with change in one area that provides data or actions to another.

4.2.4 Risk Response Planning. This step takes the results of assessment and quantification and determines what actions to take in resolving or mitigating risk. Actions may vary from simple periodic monitoring on a specific timetable, to transfer of the risk to another organization where the risk might be more tolerable.

4.2.5 Risk Monitoring & Control. While it is possible to eliminate risk, that result is generally cost or time prohibitive, and the more acceptable action is acceptance with monitoring until the severity either becomes acceptable (Considered to be within project control) or unacceptable (Severity is such that the project must terminate or make major changes to accommodate the risk).

Checklist for Identifying and Evaluating Risk

___Identify Risk Assessment Team.
___Create a Risk Assessment Plan.
___Identify potential risks.
___Assess risks.
___Quantify & rate the level of risk.
___Determine appropriate response to identified risks.
___Execute Risk Monitoring.

Figure 4.2.5-1. Checklist for Identifying & Evaluating Risk

1. 5. THE DECISION-MAKING PROCESS

Architecture development occurs for a purpose. There are different purposes for different requirements. However, there is one common purpose that should be part of each development project—collecting, organizing, and using architectural data for analysis in support of the decision-making process.

Previous sections have spoken to the collection and organization of data; the use of the DoDAF Meta-model to aid in understanding how to collect data that others can use later; and how to visualize the data to make it easier for managers at all levels to understand what has been collected, and how it can be used. Both DoDAF Volumes 1 & 2, and the DoDAF Journal provide a wide range of examples, presentation formats, and other assistance is making valuable use of the data. That is the extent of DoDAF use since it was created to assist in developing the architectures.

This section provides a brief look at what happens when the architecture development is complete—from a data perspective—and a few tips on how to use the architecture in the decision-making process. The next section, Section 6, *Linking Architectures*, also discusses reuse of architectural data among architectures that have common interfaces (i.e., some part of an architecture that has an input, impact or, or is benefited by another architecture).

Decisions are made by managers at all levels of the organization. Each level has a broad understanding of their processes and requirements, but a varying understanding of the technical aspects of architecture development. That means gaining approval of a project solution that has undergone architectural description creation needs to find ways to effectively provide information to the decision-maker that facilitates gaining the desired approvals.

Two things are important here. First, every organization has preferred ways of presenting briefings and presentations to senior leaders. Presenters have evaluated what type of data, the depth of the data, the technical or non-technical aspects of the data, and the length of presentation that a senior leader prefers. All of this type of information is generally available to the architect and the team for use during and following the development effort.

Second, with the change is emphasis in DoDAF V2.0 from a product-centered approach to a data-centered approach, it is much easier to link the presentation style and form desired by the senior leader(s) to the data available from the analysis for their decision-making effort. An architect, in deciding what models and views are needed to collect, organize, and understand the architectural data, also needs to be aware of how he/she will present that data to varying levels of senior management.

A typical scenario may involve providing technical presentations, with more formal DoDAF-described or user-described (Fit-for-Purpose) views that convey the technical direction and information needed for their technical purposes. A simpler, more briefing-

oriented presentation, containing graphical views, such as dashboard, composite or fusion views, using architectural data, or simple spreadsheet matrices to show relationships among activities and systems or services, for example, may impress the non-technical leader. Both presentations contain the same data, but the views are different to accommodate differing levels of technical comprehension in senior leaders.

Both Volume 1 and Volume 2 contain information on the wide range of models and views available to the architect and the team as examples of how to present architectural data. While they are examples, they do contain models that have been developed over several years, and are widely used and understood.

6. LINKING ARCHITECTURES

Most architectural descriptions provide information on a section of the office or organization. As we have already seen, however, architectural descriptions have an impact above and below their level of activity, and also impact stakeholders in a 360-degree circle outside that same level of activity. Suppose, for a moment, that all of the architectural descriptions created within the same organization were done completely independent of each other, were not subject to sets of common rules or terminology, and were simply allowed to 'evolve' as they were created.

Logically, there would probably be a small sub-set of the terms that would be common to two or more architectural descriptions, and a percentage of those terms would have similar definitions. However, that may leave a very large universe of terms and activities that do not match, even if they are intended to do so. Chaos generally results from those situations, and it is chaos that architectural description definition is intended to prevent within the enterprise.

In DoDAF V2.0, the DoDAF meta-model (DM2) was created to solve, at least partially, two major problems that prevent full use of architectural descriptions in the organization. First, DM2 provides a means by which solution architectures that need to 'speak' with each other across the 'seams' of the architectures can do so, and be understood on both sides of the seam. Second, DM2 gives architects a 'language' they can use, extend, and further define in their individual efforts, while facilitating reuse of existing data that is appropriate to 'adopt' for a particular solution architecture in its architectural description. Let's look briefly at some examples of how linkages can be facilitated through DM2.

The first example involves financial transactions. In many if not most financial operations, there are accounts receivable, accounts payable, and general ledger sections or offices, each with their own procedures for processing their part of the cash flow transactions. The general ledger serves as the anchor of activity, receiving payments-going-out, or payments-coming-in transactions, reviews them, and submits them for entry into the financial ledgers of the organization. The workflow is often similar to what is seen in **Figure 6.1-1** below. When a common set of terms is used, the processing goes smoothly, the payments are received/made, the entries reviewed, and the books balanced.

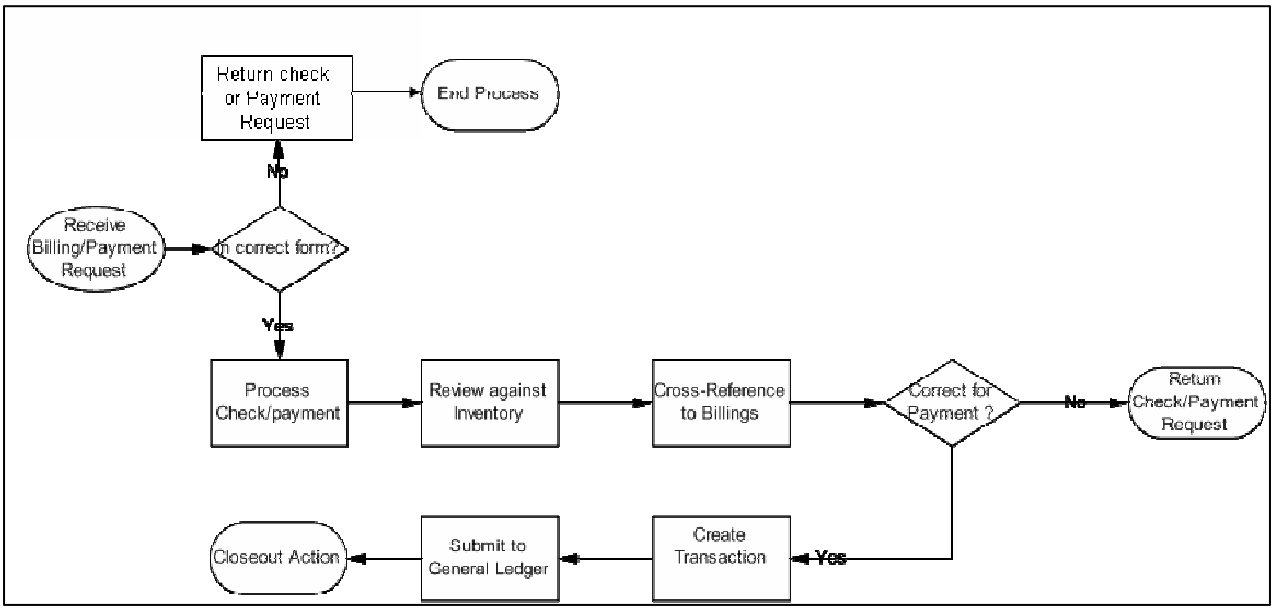

Figure 6.1-1: Workflow Diagram (OV-6c) or Check/Payment Processing

However, when there are no common terms used, the process becomes chaos. As shown in **Figure 6.1-2**, the two high-level processes for receivables and payables both have exit sub-processes to the General Ledger, but those exit points are described very differently. That means someone (Read that general ledger clerk or analyst) has to take each transaction and decide what to do next.

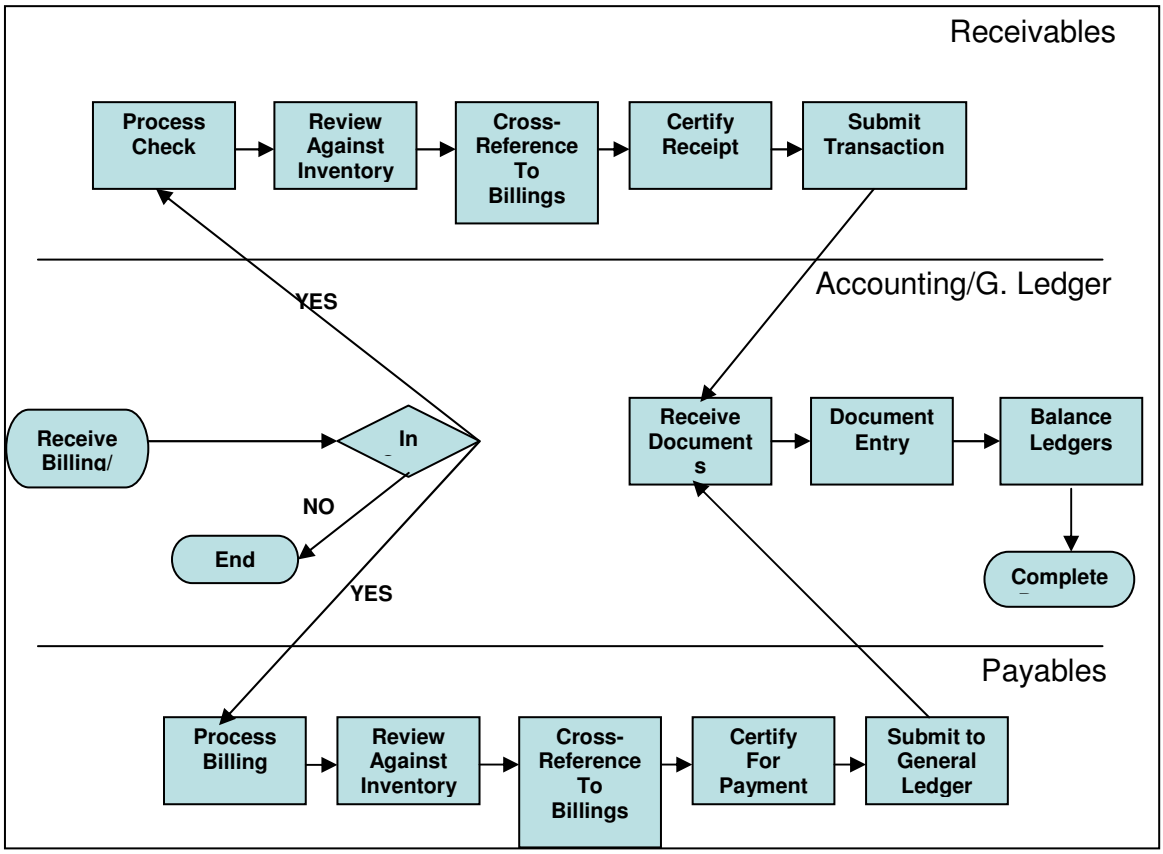

Figure 6.1-2: Consolidated General Ledger Entry Process

While the step "Submit to General Ledger is clear, the transaction "Submit Transaction" requires further explanation. In a simple workflow, such as this view, the sense that both mean the same thing is obvious, but in more complex transaction workflows, that result may not be as clear. When both aspects of the workflow have the same label for the process, then there is clarity in purpose and expected result. This same type of clarity is required in all views that cross into other processes, impact on processes external to the current architecture, or on information/processes that pass data into the current architecture for execution.

The DoDAF meta-model has been created to ensure that data is used consistently within and across architectural views, and that terms utilized in any part of the architectural description are understood in the same way wherever, and whenever they are used in architecture development.

Team members need to refer to Volume 2 of DoDAF for the structures needed to create, describe, and organize architectural data, and the methods to be utilized in collecting and organizing the data in databases, repositories, and architecture tools.

7. ARCHITECTURE MAINTENANCE AND UPDATE

When completed, an architectural description continues to be useful in analysis and decision-making and, throughout the life cycle of the process is a key reference document—an asset—of the organization. The architectural description documents how the process was built, how it works, and how it inter-relates with other processes to achieve a successful mission, operation, or business function.

Over time, as further change is needed, the architectural description is very similar to its cousin, the blueprint of a structure—it documents how a process was built, what changes were made since the original creation of the description, and what links must be considered when making changes. Every major change made in a process, and its facilitating technology and other resources should be documents by updating the architectural description, its views and viewpoints, and its presentation graphics. That way, the blueprint of the organization—its architecture—remains current, valid, and useful as it serves to aid the process owner and a new team of architects in designing and executing change.

Architectural descriptions should be reviewed at least annually, and even more frequently as change occurs. Having an up-to-date architecture provides both the process owner, and other stakeholders with factual information on their processes, expectations, requirements, and objectives. Having this information at hand assists the architect and architecture development team to clearly understand what represents the ‘baseline’, what data and other structures already exist that will be needed in a new, future state architecture, and how all these piece parts fit together in a unified enterprise.

8. WHAT DO I DO NEXT?

The answer to that question is perhaps the simplest to answer in the Guide. Follow these steps:

1- Reread this book and discuss the major parts with the architect leading your team. Find out what part you will play in the development effort.

2- Go to the sections of Volume I and Volume 2 that pertain to the overviews of those volumes and the specific areas in which you will work. Become familiar with those parts of both volumes, and the references that are cited. Collect the electronic versions of the references into a small library on your computer for later use.

3- Look over the statement of work or requirement document provided for the architectural development and match those requirements and tasks to your expected duties and, in turn, to the relevant sections of this guide and DoDAF.

Creating views from models, and generating architectural descriptions is not an exact task. It takes patience, willingness to learn from mistakes, and a desire to explore in addition to documenting present and future states. However, the data-centered approach is designed to ensure that a first-time effort is just as accurate as one produced by experienced architects and teams—it can be so if you follow the steps, learn where to find the information you need, and accurately document those findings.

Appendix A: Frequently Asked Questions

The most commonly asked:

1. Do all models/views need to be created?
No, the models and views needed depend on the purpose, scope, and the policy and regulatory requirements that guide the development.

2. What is the minimum set of models/views that need to be created?
DoDAF does not prescribe any mandatory models or views. Policy, such as CJCSI 6212 and 3170, indicates the specific requirements for systems/services developed under those policies. In any architectural development, there may be multiple stakeholders, with unique and overlapping requirements for models and views. What DoDAF does prescribe is the data that is necessary for that model or view.

3. Am I required to use specific tools for developing architectural descriptions?
No. DoDAF does not prescribe particular toolsets for developing architectural descriptions. Any toolset capable of importing and exporting the Physical Exchange Specification (PES) Data described in DoDAF Volume 3 can be used.

4. What about methodologies? Is there a required methodology?
DoDAF is methodology, technique, and notation agnostic. That means architects and teams can use the methodology, supporting techniques, or notations that are prescribed by their own leaders. DoDAF, Volume 1 provides a notional methodology to ensure that any selected methodology can be compared to it for completeness of the steps needed from planning to execution.

Similarly, architects and teams can use varying techniques (SADT, IDEF, UML, etc.) or notations (BPMN, etc.) that will facilitate their desired end results, such as process improvement, systems or services development, or a combination of both.

5. Where do I go for more information?
The DoDAF Journal is the first place at https://www.us.army.mil/suite/page/454707 along with the volumes of DoDAF. At the appendices of DoDAF Volume 1, there is a bibliography of know sources of information. The DoDAF Journal will expand over time to include more lessons learned, examples, and other resources to assist development teams.

This page left intentionally blank

Appendix B: Acronyms

This is the integrated DoDAF V2.0 acronyms and their definitions. Some have more than one definition depending on their usage; they could have a specific meanings in Architecture as well as generic English language usage.

Term	Definition
A	
A&T	Acquisition and Technology
AIS	Automated Information System
ASD (C3I)	Assistant Secretary of Defense for Command, Control, Communications, and Computer Systems Directorate, Joint Staff
AV	All Viewpoint
B	
BMM	Business Motivation Model
BPMN	Business Process Modeling Notation
BPR	Business Process Reengineering
BRM	Business Reference Model
BT	Business Transformation
BTA	Business Transportation Agency
C	
CADM	Core Architecture Data Model
C3I	Command, Control, Communications, and Intelligence
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance
CDM	Conceptual Data Model
CCB	Configuration Control Board
CDM	Conceptual Data Model
CI	Configuration Item
CIO	Chief Information Officer
CJCS	Chairman of the Joint Chiefs of Staff
CJCSI	Chairman of the Joint Chiefs of Staff Instruction
CM	Configuration Management
COI	Communities of Interest
CPIC	Capital Planning and Investment Control
CPM	Capability Portfolio Management
CRM	Consolidated Reference Model
CV	Capability Viewpoint
CWID	Coalition Warrior Interoperability Demonstration
D	
DAES	DoD Architecture Enterprise Services
DARS	Department of Defense Architecture Registry System
DAS	Defense Acquisition System
DDMS	DoD Discovery Metadata Specification
DFD	Data Flow Diagrams
DIE	DoD Information Enterprise
DIEA	DoD Information Enterprise Architecture
DISR	DoD Information Technology Standards Registry
DIV	Data & Information Viewpoint
DMM	DoDAF Meta-model

Term	Definition
DoD	Department of Defense
DoD EA	DoD Enterprise Architecture
DoD EA BRM	DoD EA Business Reference Model
DoDAF	DoD Architecture Framework
DODI	Department of Defense Instruction
DOTMLPF	Doctrine, Organization, Training, Material, Leadership and education, Personnel, and Facilities
DRM	Data Reference Model
E	
EA	Enterprise Architecture
EAAF	Enterprise Architecture Assessment Framework
EAMMF	Enterprise Architecture Management Maturity Framework
EIA	Electronic Industries Alliance
ES	Enterprise Services
F	
FEA	Federal Enterprise Architecture
FEA RM	Federal Enterprise Architecture Reference Model
FEAF	Federal Enterprise Architecture Framework
FIPS	Federal Information Processing Standard
G	
GAES	GIG Architecture Enterprise Services
GAO	Government Accountability Office
GIG	Global Information Grid
H	
I	
IDEF	Integration Definition (IEEE)
IDEF0	Integration Definition for Activity Modeling
IDEF1X	Integration Definition for Data Modeling
IDEF3	Integration Definition for Process Description Capture
ISO	International Standards Organization
IT	Information Technology
JCA	Joint Capability Area
JCIDS	Joint Capabilities Integration and Development System
JCS	Joint Chief of Staff
JFCOM	US Joint Forces Command
JP	Joint Publication
L	
LDM	Logical Data Model
LOB	Line of Business
M	
M3	MODAF Meta Model
MOD	Ministry of Defense (UK)
MODAF	Ministry of Defense Architecture Framework
N	
NAF	NATO Architecture Framework
NC	Net-Centric Net Centric (JCA)
NCDS	Net-centric Data Strategy
NCDSWG	Net Centric Data Strategy Working Group
NCE	Net-Centric Environment
NCES	Net-Centric Enterprise Services
NCO	Net-Centric Operations

Term	Definition
NCOW	Net-Centric Operations and Warfare
NCOW RM	Net-Centric Operations and Warfare Reference Model
NII	Networks and Information Integration
NSS	National Security Systems
O	
OASD	Office of the Assistant Secretary of Defense
OMB	Office of Management and Budget
OOAD	Object-Oriented Analysis & Design Technique
OSD	Office of the Secretary of Defense
OSD(NII) A&I	Office of the Secretary of Defense Networks and Information Integration Architectures and Integration
OV	Operational Viewpoint
P	
PDA	Personal Digital Assistant
PDCA	Plan, Do, Check, Act
PDM	Physical Data Model
PEOs	Program Executive Office
PFD	Process Flow Diagram
PfM	Portfolio Management
PPBE	Planning, Programming, Budgeting, and Execution
PRM	Performance Reference Model
PTD	Process Task Dependency
PV	Project Viewpoint
Q	
QA	Quality Assurance
QC	Quality Control
R	
RM	Reference Model
S	
SADT	Structured Analysis and Design Technique
SE	Systems Engineering
SECDEF	Secretary of Defense
SEI	Systems Engineering Institute
SeV	Systems Viewpoint
SLC	Shelf-Life Code
SOA	Service-Oriented Architecture
SRM	Service Component Reference Model
SrV	Service Viewpoint
StdV	Standards Viewpoint
SUMO	Suggested Upper Merged Ontology
SysML	Systems Modeling Language
T	
TA	Technical Architecture?? Tiered Accountability
TAFIM	Technical Architecture for Information management
TOGAF	The Open Group Architecture Framework
TRM	Technical Reference Model
TSAT	Transformational Communications Satellite
TTP	Tactics, Techniques, and Procedures
TWG	Technical Working Groups
U	
UML	Unified Modeling Language

Term	Definition
URL	Uniform Resource Locator
USD(AT&L)	Under Secretary of Defense for Acquisition, Technology & Logistics
USJFCOM	United States Joint Forces Command
V	
V&V	Validation & Verification
W	
WBS	Work Breakdown Structure
X	
XSD	XML Schema Definition

Appendix C: The Quick Locator

Defense Knowledge Online (DKO): https://www.us.army.mil/suite/portal/index.jsp

DoD Architecture Registry System (DARS): https://dars1.army.mil

DoD Metadata Registry (DMM): http://metadata.dod.mil

DoDAF Journal: https://www.us.army.mil/suite/page/454707

Federal Enterprise Architecture (FEA): http://www.whitehouse.gov/omb/egov/a-1-fea.html

Zachman Framework: http://zachmaninternational.com/index.php/the-zachman-framework/26-articles/13-the-zachman-framework-a-concise-definition

This page left intentionally blank

Appendix D: Bibliography

This section provides a listing of directives, instructions, manuals, and other guidance routinely used in development and management of architectures in DoD.

ANSI/GEIA Standard EIA 649-A National Consensus Standard for Configuration Management American National Standards Institute. This standard is available at: http://www.techstreet.com/cgi-bin/detail?product_id=1160265

Chairman of the Joint Chiefs of Staff (CJCS) Instruction 3170.01E, Joint Capabilities Integration and Development System (JCIDS), 11 May 2005. A copy of the current version of the instruction and its accompanying Manual can be found at: https://acc.dau.mil/CommunityBrowser.aspx?id=42776

Department of Defense Directive (DoDD) 5000.1, The Defense Acquisition System, 12 May 2003 (certified current as of November 20, 2007). A current copy of the directive can be found at: https://akss.dau.mil/dag/DoD5000.asp?view=document&doc=2

Department of Defense Directive (DoDD) 8115.01, Information Technology Portfolio Management, October 10, 2005. Office of the Assistant Secretary of Defense (Networks & Information Integration) (NII)/DoD Chief Information Officer (DoD CIO). The latest copy of this directive can be found at: http://www.dtic.mil/whs/directives/corres/rtf/811501x.rtf

Department of Defense Instruction 4630.8, Procedures for Interoperability and Supportability of Information Technology (IT) and National Security Systems (NSS) 30 June 2004. Office of the Assistant Secretary of Defense (Networks & Information Integration) (NII)/ DoD Chief Information Officer (DoD CIO). The current version is found at: www.dtic.mil/whs/directives/corres/pdf/463008p.pdf

Department of Defense Instruction (DoDI) 5000.2., Operation of the Defense Acquisition System. (2003) Under-Secretary of Defense (Acquisition, technology & Logistics) (OUSD AT&L). A current copy of this document can be found at: https://akss.dau.mil/dag/DoD5000.asp?view=document&doc=2

Department of Defense Manual 8210-11-M, DoD Architecture Federation Manual, dated (DRAFT) XX-XX-200X Office of the Assistant Secretary of Defense (Networks and Information Integration) (NII)/ DoD Chief Information Officer (DoD CIO).

Department of Defense Net-Centric Data Strategy, 9 May, 2003. Office of the Assistant Secretary of Defense (Networks & Information Integration) (NII)/DoD Chief Information Officer (DoD CIO).

DoD Acquisition Guidebook. Office of the Under-Secretary for Acquisition, Technology & Logistics (AT&L). A current copy of the Guidebook can be found at: https://akss.dau.mil/dag/DoD5000.asp?view=document&doc=2

Federal Enterprise Architecture (FEA). Executive Office of the President, Office of Management and Budget E-Gov Initiative. The current version of the FEA and its associated reference models can be found at: http://www.whitehouse.gov/omb/egov/fea

Federal Enterprise Architecture - Business Reference Model. The current version can be found at: http://www.whitehouse.gov/omb/egov/a-3-brm.html

Federal Enterprise Architecture Consolidated Reference Model Version 2.3. Executive office of the President, Office of Management and Budget (OMB). A current version can be found at: http://www.whitehouse.gov/omb/assets/fea_docs/FEA_CRM_v23_Final_Oct_2007_Revised.pdf

Federal Enterprise Architecture Program: Enterprise Architecture Assessment Framework, version 2.2, October 2007. Executive Office of the President. Office of Management and Budget. The current version can be found at: http://www.whitehouse.gov/omb/egov/a-2-EAAssessment.html.

Federal Enterprise Architecture Records Management Profile, Version 1.0, December 15, 2005. Executive Office of the President. Office of Management and Budget. A current version of the profile can be found here: http://www.cio.gov/documents/RM_Profile_v1.pdf

Global Information Grid (GIG) Architecture Federation Strategy, 1 August 2007. Office of the Assistant Secretary of Defense (Networks & Information Integration) (NII)/DoD Chief Information Officer (DoD CIO).

Information Sharing Environment Enterprise Architecture Framework (DRAFT) June, 2008. Office of the Program Manager, Information Sharing Environment.

Integrated Defense Acquisition, Technology, & Logistics Life Cycle Management Framework (2005). Defense Acquisition University, Ft. Belvoir, VA. A current copy of the chart is found at: http://www.dau.mil/pubs/IDA/IDA_04.aspx

International Standards Organization (ISO) (2007*). Systems and Software Engineering – Architectural Description* ISO/IEC WD1 42010 7 Apr 2007 (Requires purchase through the ISO.ORG website)

Intelligence Reform and Terrorism Prevention Act of 2004 (IRTPA), PL 108-458 (December 17, 2004)

Office of Management and Budget (OMB) Circular-A-130, Management of Federal Information Resources, February 8, 1996. Executive Office of the President. Office of Management and Budget. The current version can be found at: http://www.whitehouse.gov/omb/circulars/a130/a130trans4.html#2

United States Government Accountability Office (GAO) Report: DoD Business Systems Modernization: Long-standing Weaknesses in Enterprise Architecture Development Need to Be Addressed, July 2005, GAO-05-702. A copy of the report is available at: http://www.gao.gov/new.items/d05702.pdf

United States Government Accountability Office (GAO) Report: Framework for Assessing and Improving Enterprise Architecture Management, version 1.1, April 2003, GAO-03-584G. A copy of the report is available at: www.gao.gov/new.items/d03584g.pdf

Bibliography

Dinoli, D. (2008) *Enterprise Architecture A to Z: Frameworks, Business Process Modeling, SOA, and Infrastructure Technology*. Auerbach Publishing. 504pp.

Grigoriu, A. (2007) An *Enterprise Architecture Framework*. Privately Published 226.pp.

McGovern, J Ambler, S Stevens, M. E., Linn, J, Sharan, V & Jo, E. K. (2004) A *Practical Guide to Enterprise Architecture*. New Jersey: Prentice-Hall. 306pp.

Ministry of Defense (UK) (2007). *Ministry of Defense Architecture Framework (MODAF).* The latest electronic version of the MODAF can be found at the MODAF Website: http://www.modaf.org.uk/ . An offline (PDF) version can be accessed here for downloading: 20081001-MODAF_Handbook_V1_2_003_Adobe6-U.pdf [5.51MB]

North Atlantic Treaty Organization (NATO) (2006). *NATO Architecture Framework (NAF) for C3 Systems. Brussels, Belgium.* The latest version of the NAF, and other associated documents can be found at the NATO Website http://194.7.80.153/website/book.asp?menuid=15&vs=0&page=volume2%2Fch02s02.html (Requires User Registration)

Ross, J. W., Weill, P., & Robertson, D. C. (2006). *Enterprise Architecture as Strategy*. Boston, MA: Harvard Business School press. 235pp.

Ross, R. (2005). *Business Rule Concepts* (2nd. Ed.) Business Rules Forum. 135pp.

Ross, R. (2003). *Principles of the Business Rule Approach*. Boston, MA: Addison-Wesley Publishing. 372pp.

The Open Group (2007). *The Open Group Architecture Framework (TOGAF), Version 8.1.1* The current version can be found at: http://www.opengroup.org/architecture/togaf/

Tieso, J. & Thornburg, C. (2007) *Managing the Lifecycle of Net-centric Architectures.* Proceedings of the 13th International Command & Control Research & Technology Symposium. Bellevue, WA. June 2008.

Zachman, J. *Zachman Framework*. © Zachman International. The Zachman Framework can be found at the Zachman International Website: http://zachmaninternational.com/index.php/the-zachman-framework/26-articles/13-the-zachman-framework-a-concise-definition